AUTO-DOCS

VIT lions

Table of Content

1.Topic of the Project

2.Problem Statement/objective/Scope of the project

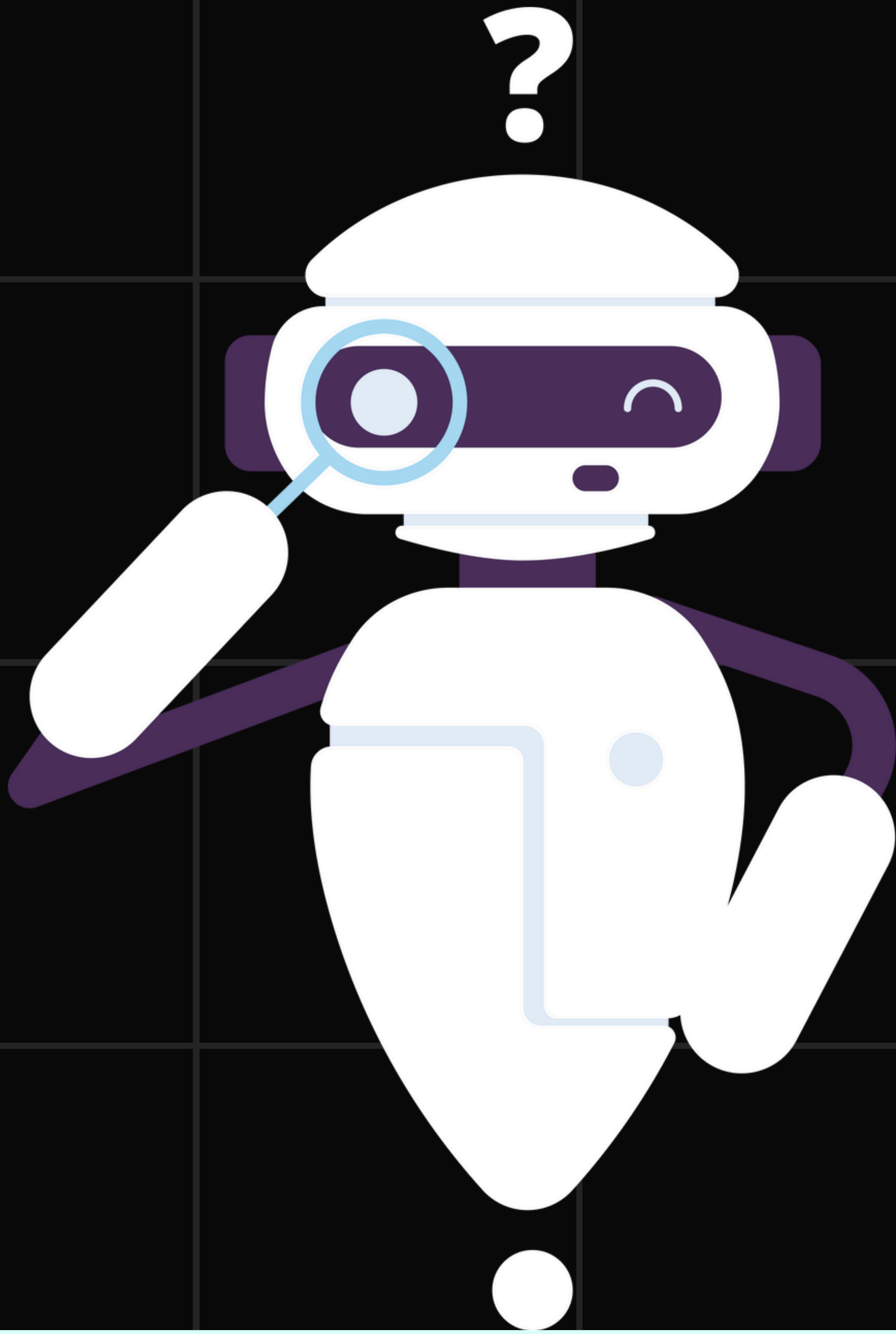
3. Solution Proposed

4.Hardware / Software Requirement

5. Working Architecture

6. Usability / Applications

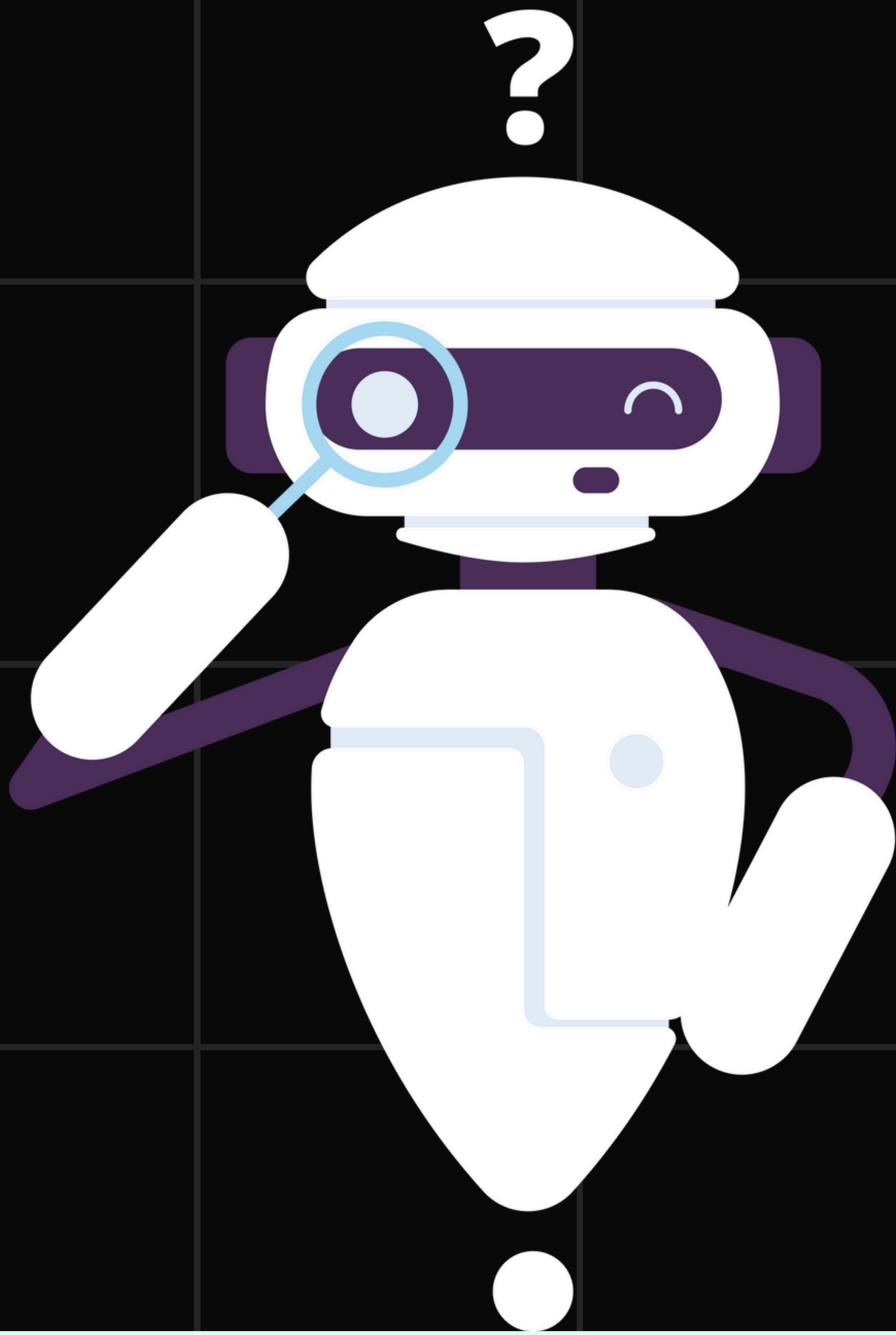
PROBLEM STATEMENT



Developers often struggle with documenting projects and generating helpful context for large codebases—especially during collaboration, onboarding, or when using AI tools. Manually writing documentation, README files, or preparing for LLM-based assistance is time-consuming and error-prone as:-

1. Manual documentation is time-consuming and often neglected by developers due to tight deadlines or lack of clarity.
2. Large and complex codebases are hard to understand, especially without proper context or structured documentation.
3. Onboarding new developers becomes difficult when projects lack consistent README files or architectural overviews.
4. LLMs (like ChatGPT or Gemini) require well-structured input, but developers struggle to prepare clean project context manually.
5. Maintaining up-to-date documentation is error-prone and often forgotten during rapid development cycles.
6. Lack of visual aids like flowcharts makes understanding project logic and structure more difficult, especially for non-authors.

SOLUTION PROPOSED



We present an Auto-Docs VS Code Extension that provides:

Project to File Conversion

– Converts the entire workspace/project into a single .txt file with code and structure, for LLMs or documentation purposes.

Active Window to File

– Saves the currently opened file's content to a .txt file for quick access or AI input.

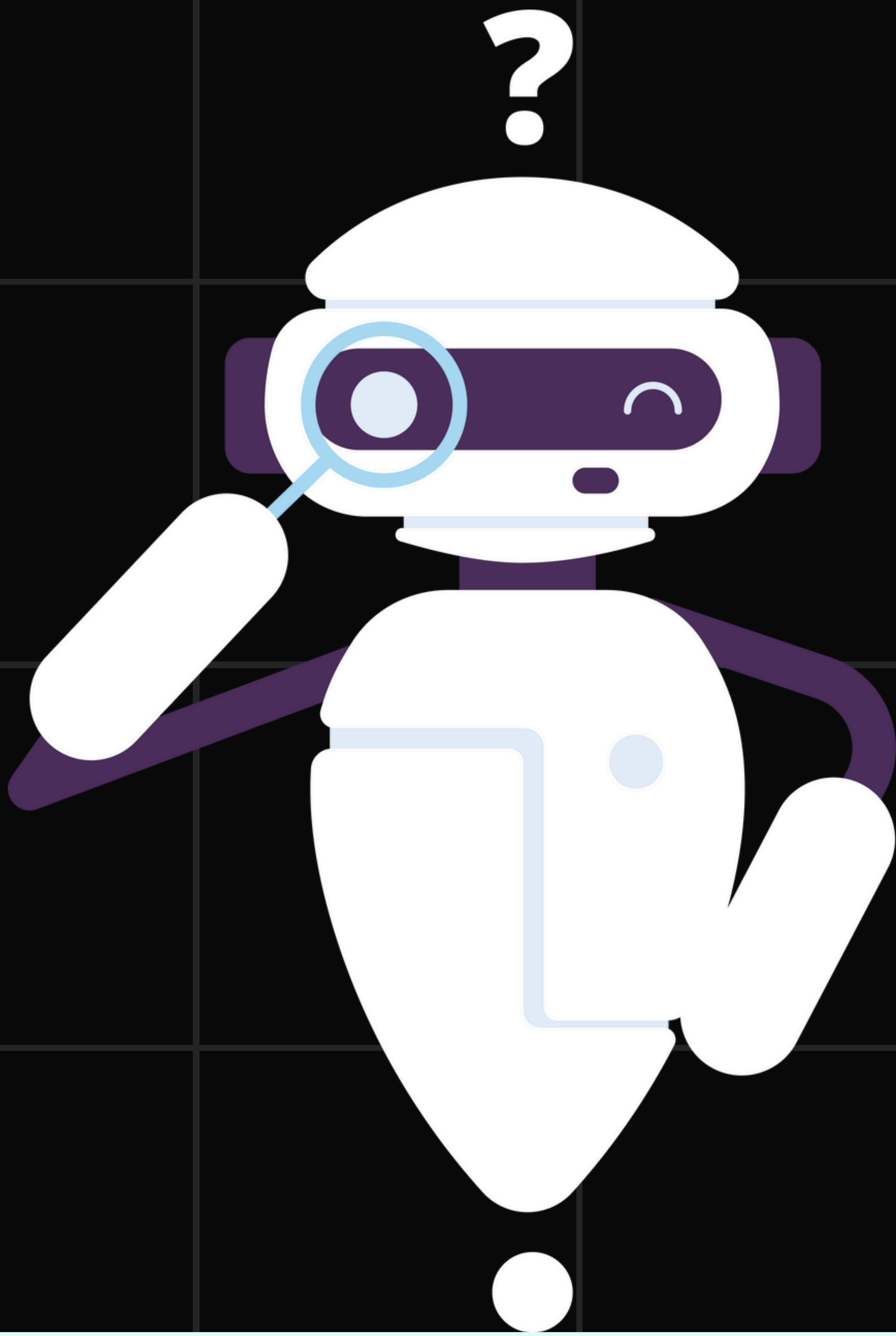
README.md Generation (via Gemini API)

– Auto-generates a professional README.md using real-time codebase parsing + Google Gemini LLM.

Flowchart Generation (Planned/Integrated)

– Generates Mermaid-based flowcharts using code context for visualizing application structure.

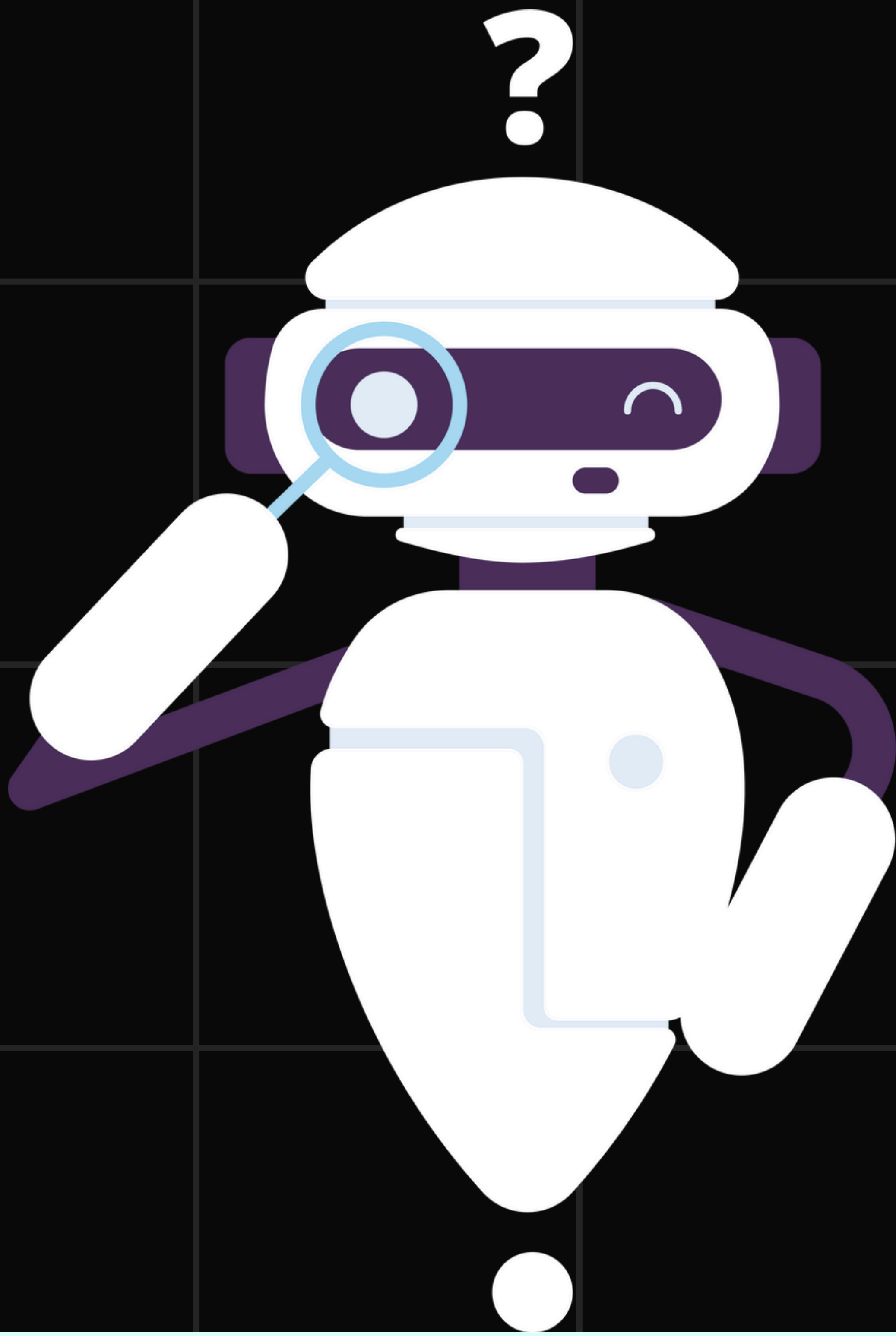
SOFTWARE REQUIRED

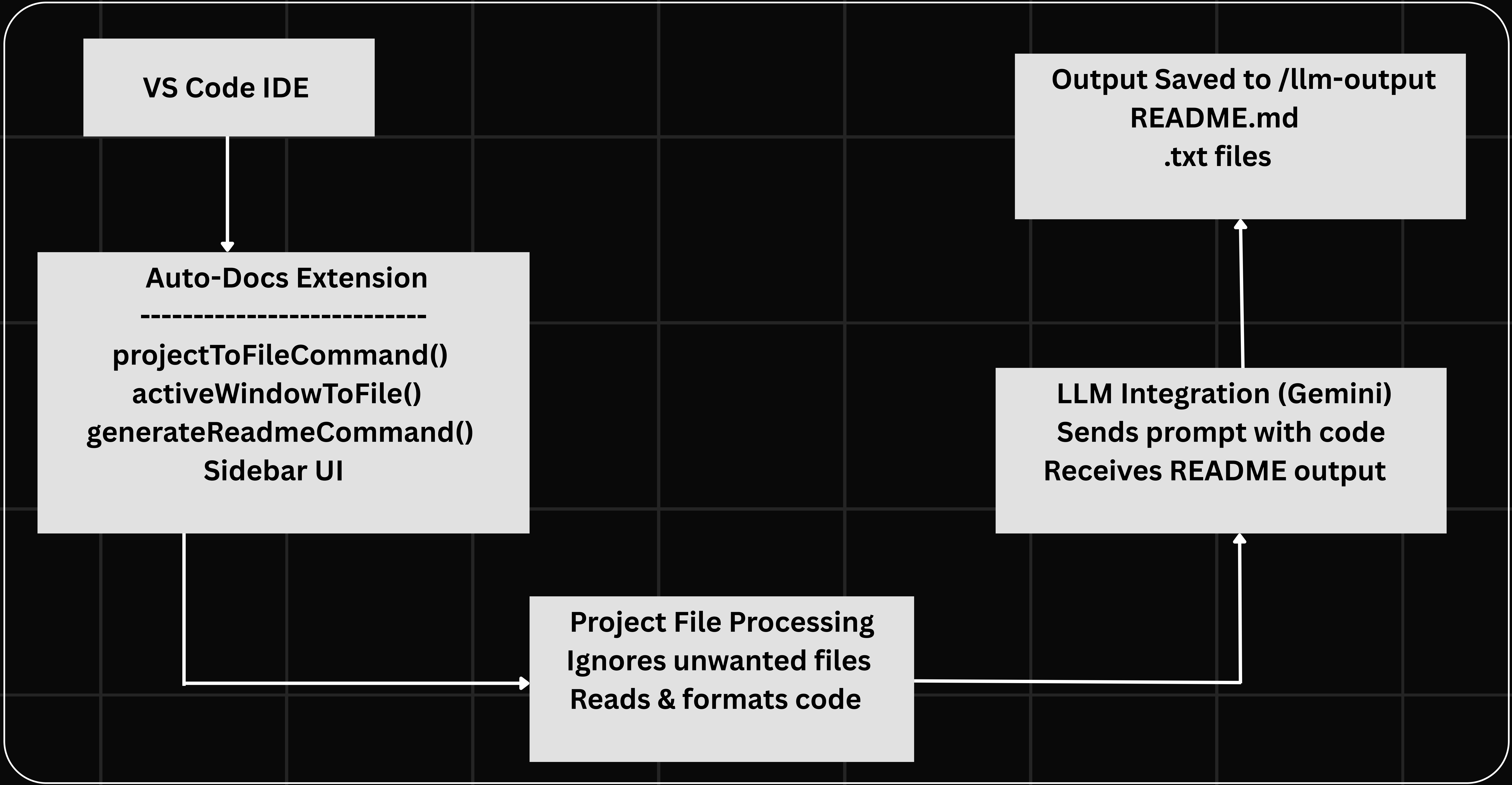


🔧 Software:

- VS Code
- Node.js (for extension dev & runtime)
- Google Gemini API (for LLM-powered README generation)
- Mermaid.js (for flowchart rendering)
- Optional: Git, Clerk API, Prisma (if integrated in final version)

WORKING ARCHITECTURE





VS Code IDE

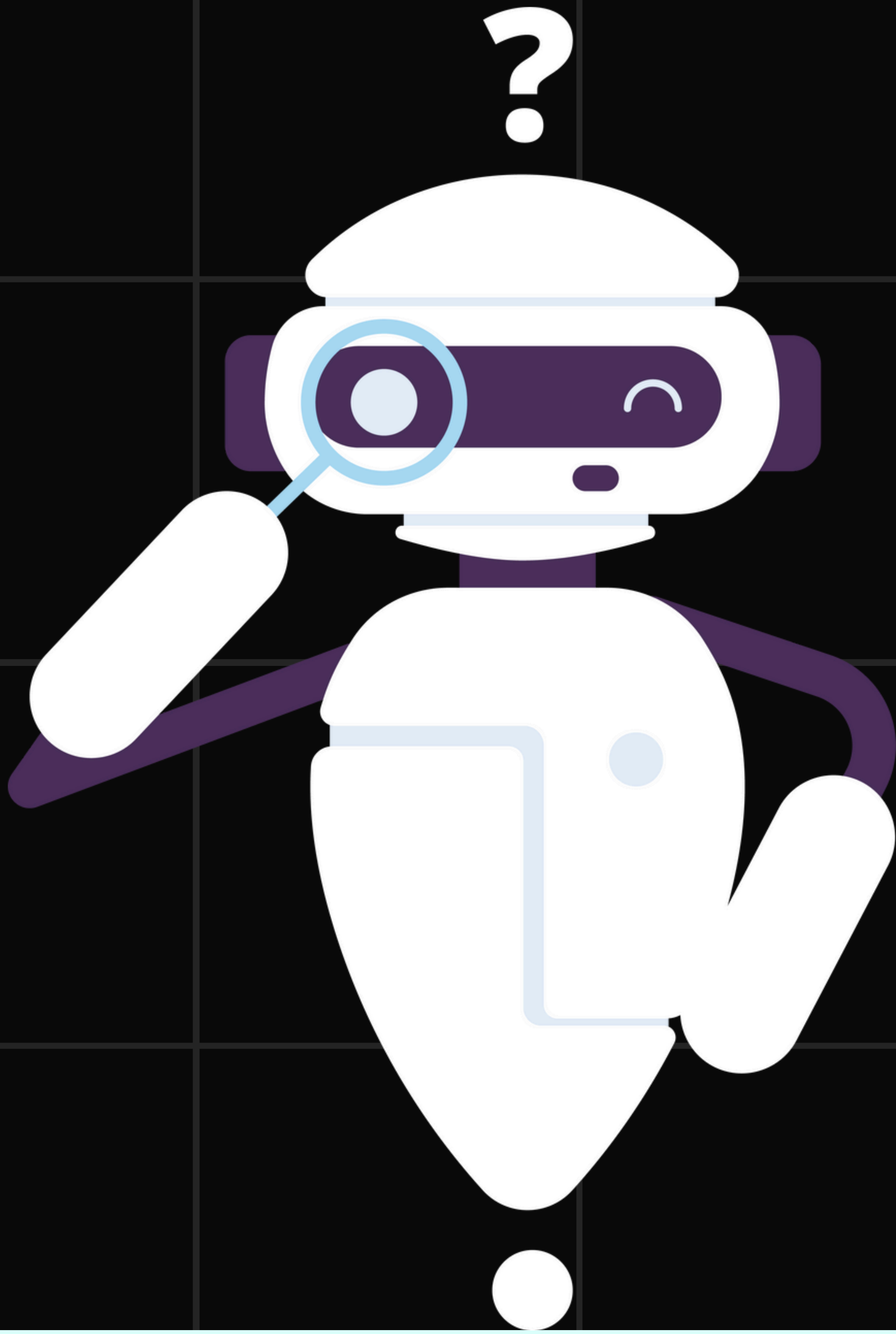
Auto-Docs Extension
----
projectToFileCommand()
activeWindowToFile()
generateReadmeCommand()
Sidebar UI

Project File Processing
Ignores unwanted files
Reads & formats code

LLM Integration (Gemini)
Sends prompt with code
Receives README output

Output Saved to /llm-output
README.md
.txt files

Usability & Application



Developer Use-Cases:

- Instantly generate documentation for unfamiliar or legacy projects
- Feed structured code context to LLMs for code explanation, test writing, or debugging
- Speed up README writing for open-source or hackathon submissions

Applicable In:

- Hackathons (fast prototyping)
- Startups and open-source projects
- Developer onboarding & internal tools
- LLM-based dev tools (copilots, code reviewers)
- Educational tools for teaching code structure

Thankyou